

# JP2000299863

Publication Title:

## IMAGE COMPRESSING DEVICE

Abstract:

Abstract of JP2000299863

**PROBLEM TO BE SOLVED:** To reduce the size of an off-chip memory and the number of input- output operations between off-chip memories by filtering plural rows of an image at a time until the whole image is decomposed into plural subbands. **SOLUTION:** A chip 22 is provided with a 1st stage 30 of a subband decomposition filter. The chip 22 reads N-row windows from an off-chip memory 24, the stage 30 generates plural subbands HH, HL, LH and LL, and respective subbands are generated for S-rows at a time in this way. Here, the S is defined as a positive integer, and  $S < N$  is defined. For instance, in the case of  $N=10$  and also  $S=1$ , a one-row transformation coefficient is generated about the respective subbands HH, HL, LH and LL due to the decomposition of a ten-row window. Further, a system 20 is provided with a 2nd machine 44 having a decoder and reconstructs an image from an embedded bit stream.

Data supplied from the esp@cenet database - Worldwide

---

Courtesy of <http://v3.espacenet.com>

(19)日本国特許庁 (JP)

## (12) 公開特許公報 (A)

(11)特許出願公開番号  
特開2000-299863  
(P2000-299863A)

(43)公開日 平成12年10月24日 (2000.10.24)

(51) Int.Cl.<sup>7</sup>  
 H 04 N 7/30  
 H 03 M 7/30  
 H 04 N 1/411  
 // G 06 F 5/00

識別記号

F I  
 H 04 N 7/133  
 H 03 M 7/30  
 H 04 N 1/411  
 G 06 F 5/00

テ-マコ-ド<sup>8</sup> (参考)  
 Z  
 A  
 H

審査請求 未請求 請求項の数 1. O.L. (全 16 頁)

(21)出願番号 特願2000-62952(P2000-62952)  
 (22)出願日 平成12年3月8日 (2000.3.8)  
 (31)優先権主張番号 09/267246  
 (32)優先日 平成11年3月12日 (1999.3.12)  
 (33)優先権主張国 米国 (US)

(71)出願人 398038580  
 ヒューレット・パッカード・カンパニー  
 HEWLETT-PACKARD COMPANY  
 アメリカ合衆国カリフォルニア州パロアルト ハノーバー・ストリート 3000  
 (72)発明者 デーヴィッド・エス・トーブマン  
 オーストラリア、2073、ニューサウスウェールズ、ピンブル、ピンブル・アヴェニュー  
 39  
 (74)代理人 100081721  
 弁理士 岡田 次生

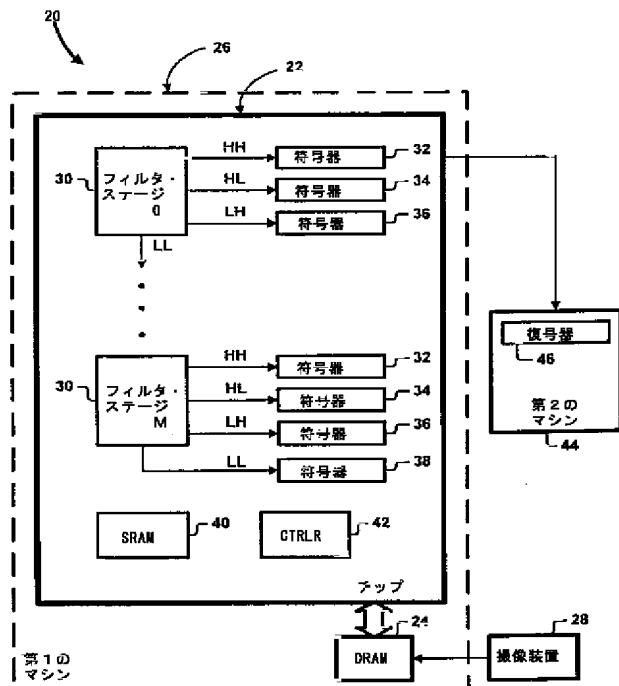
最終頁に続く

## (54)【発明の名称】 画像圧縮装置

## (57)【要約】

【課題】オフチップ・メモリのサイズを削減し、チップおよびオフチップ・メモリ間の入出力操作数を削減する。

【解決手段】画像のサブバンド分解が、画像をN行ウィンドウずつ順次フィルタリングすることにより実行される。第1のフィルタ・ステージでは、垂直フィルタがそれぞれのN行ウィンドウに対して高域および低域フィルタリングを実行する。次に、それぞれの垂直フィルタの出力に対して水平フィルタリングが実行される。フィルタ・ステージが提供する最低サブバンドが、追加のフィルタ・ステージにより帰納的に分解される。それぞれのフィルタ・ステージのこれより高いサブバンドは、符号化され、埋め込みビットストリームに配置される。ブロック符号化が実行され、それぞれのサブバンドの複数行が複数のブロックとして符号化される。サブバンドのブロックは相互に独立して符号化される。ブロック符号化は、関心領域の再構成をサポートする。



**【特許請求の範囲】**

【請求項1】K行（Kは正の整数）を含む画像を圧縮する装置であって、前記画像のN行（Nは正の整数、 $1 < N < K$ 、NはKから独立している）ウィンドウを順次提供する手段と、垂直高域フィルタおよび垂直低域フィルタを有するサブバンド分解フィルタの第1のステージであって、前記提供されたそれぞれのN行ウィンドウに対して垂直高域および垂直低域フィルタリングを実行する第1のステージを有するチップとを備え、前記画像全体が複数のサブバンドに分解されるまで、前記チップが、前記画像のN行を一度にフィルタリングするようにした画像圧縮装置。

**【発明の詳細な説明】****【0001】**

【発明の属する技術分野】この発明は、データ圧縮に関し、より具体的には、画像のサブバンド分解に関する。

**【0002】**

【従来の技術】データ圧縮は、コンピュータ上に大型のデータファイルを格納するコストを削減するため、およびコンピュータ間で大型のデータファイルを伝送する時間を短縮するためにしばしば使用される。いわゆる「変換方法」により、データは、周波数領域でデータを表す係数に変換される。係数を、データの質に重大な影響を与えることなく量子化することができ（損失圧縮）、最終的には量子化された係数からこのデータが再構成される。その後、係数の冗長部分を、再構成されたデータの質に影響を与えることなく低減または除去（無損失圧縮）することができる。

【0003】周知の変換の1つにウェーブレット(Wavelet)変換がある。ウェーブレット変換を使用して、サブバンド分解を実行し、データを階層的な多段階表現で記述する係数を生成することができる。ウェーブレット変換は、画像の圧縮および信号の分析に有効であることが立証されている。ウェーブレット変換は、新しい「JPEG-2000」標準の変換として提案されている。

【0004】サブバンド分解は、典型的には、画像全体をメモリに格納し、画像全体の高域および低域フィルタリングを一方の方向（たとえば、垂直方向）について実行し、次にフィルタリングされた画像全体の高域および低域フィルタリングを他方の方向（たとえば、水平方向）について実行する。

【0005】ウェーブレット変換の利点の中でも特に有利なのが、変換係数を階層構造で順序付け、「埋め込みビットストリーム（embedded bitstream）」で伝送できることである。埋め込みビットストリームは、ビットストリームのプレフィックスが、画質の最高可能レベルにおいて、より低いレートのデータ記述の連続体を生み出すという特性を持つ。たとえば、埋め込みビットストリームが画像データの伝送中に切り捨てられた場合、すで

に伝送された情報によって画像全体を再構成することができる。再構成された画像の画質は、伝送された情報の量によって決まる。埋め込みビットストリームが切り捨てられた場合、画質は低下するが、伝送されたビットから完全な画像が再構成される。対照的に、非埋め込み伝送の切り捨てでは、画像の一部の行を再構成できるにすぎない。

【0006】追加情報が伝送されると、再構成画像の画質は向上する。ビットストリーム全体が切り捨てられずに伝送された場合は、無損失またはほとんど無損失の画像を再構成することができる。

【0007】上記の伝送は、「画質によるプログレッシブ（progressive-by-quality）」の画像伝送と呼ばれることが多い。係数はビットプレーンにより記述され、最も有効な係数ビット（すなわち、最も重要な情報を運ぶ係数ビット）が最初に伝送される。

【0008】他の種類の伝送は、「解像度によるプログレッシブ（progressive-by-resolution）」伝送とよく呼ばれる。解像度によるプログレッシブ伝送は、画像解像度のレベルの違いに従った係数の順序付けを含む。レベルの違いは、埋め込みビットストリーム内のマーカーによって識別される。コンピュータは、このマーカーを使用してビットストリームを解析し、受信コンピュータによって指定された解像度に対応する係数のデータを伝送する。受信コンピュータは、指定された解像度に従って画像を再構成することができる。

【0009】画像の圧縮用のフィルタおよび符号器を、1つの特定用途向け集積回路（「ASIC」）すなわち「チップ」上に作製することができる。また、メモリの小さなキャッシュをそのチップ上に作製することもできる。

【0010】フィルタリングに先だって、画像全体は、DRAMのようなオフチップ・メモリに格納されるのが一般的である。画像全体を、SRAMのようなより高速なオンチップ・メモリに格納することができるが、オンチップ・メモリは、一般にオフチップ・メモリよりずっと高価である。さらに、画像が完全にオンチップ・メモリに格納されると、チップの使用は、あるサイズの画像に制限されることになる。

【0011】しかし、画像全体をオフチップ・メモリに格納することにもまだ問題はある。この場合も、メモリは、画像全体を格納できるだけの十分な大きさを持たなければならない。さらに、オフチップ・メモリは、ASICのコストと比較して高価である。メモリのサイズが増加すると、サブバンド分解を実行するコストも増加する。このことは、画像圧縮および画像再構成についても当てはまる。

【0012】たとえば、1200 dpi（1インチあたりのドット数）で、かつ1画素あたり24ビットの印刷アプリケーションでは、約380 MBを使用してページ

全体を格納する（圧縮しないで）。近い将来、解像度は、1画素あたり24ビットで2400dpiまで増加するであろう。これには、ページ全体を格納するためには、さらに多くのメモリが必要になる。メモリを増加するコストは些細に見えるかもしれないが、プリンタのように容積が大きく、かつ利益幅の低いアイテムにとって、コストの増加はきわめて重大である。

【0013】オフチップ・メモリに関わる他の問題は、入出力帯域幅である。画像のサイズおよび解像度が増加すると、コンピューターの複雑さおよびメモリ入出力帯域幅に対する制約が強まる。増加する入出力トランザクション数に対処するために入出力帯域幅を増やすことはコストがかかるが、入出力帯域幅を維持することは（画像サイズおよび解像度が増加したとする）、画像印刷やビデオ・モニタ上の画像表示のような様々なアプリケーションで、許容できない待ち時間が生じることがある。

#### 【0014】

【発明が解決しようとする課題】オフチップ・メモリのサイズを削減したいニーズがある。また、チップおよびオフチップ・メモリ間の入出力操作数を削減したいニーズがある。画像解像度および画像サイズが大きくなるにつれて、これらのニーズは大きくなっていくであろう。

【0015】さらに、画像のサイズによって制限されることなく、異なるアプリケーションに使用することができるチップを備えることが好ましい。また、チップが、埋め込みビットストリームを生成する、または埋め込みビットストリームから画像を再構成することが望ましい。

#### 【0016】

【課題を解決するための手段】これらのニーズおよび要望は、この発明の様々な側面により達成される。この発明の1つの側面に従うと、画像は、複数のサブバンドに分解される。画像のN行ウィンドウが連続的に与えられる。ここで、Nは正の整数であり、 $1 < N < K$ であり、NはKから独立している。第1のフィルタ・ステージで、垂直方向の高域および低域フィルタリングが、与えられたそれぞれのN行ウィンドウに対して実行される。したがって、画像全体が複数のサブバンドに分解されるまで、一度に画像のN行がフィルタリングされる。

【0017】この発明の他の側面に従うと、サブバンドの変換係数が、埋め込みビットストリームに符号化される。埋め込みビットストリームは、圧縮された形態で画像を表す。

【0018】この発明のまた別の側面に従うと、埋め込みビットストリームを複数の行に復号化するステップと、所与の分解レベルのサブバンドのそれについてN行のチャックを形成するステップと、N行のそれぞれのチャックに対して逆水平フィルタリングを実行するステップと、水平にフィルタリングされたチャックに対し

て逆垂直フィルタリングを実行するステップとにより、埋め込みビットストリームから画像を再構成することができる。

【0019】この発明の他の側面および利点は、この発明の原理の例を説明する図と共に、以下に示す詳細な説明から明らかになるであろう。

#### 【0020】

【発明の実施の形態】説明のための図に示すように、この発明は、大型画像を圧縮および再構成する方法および装置によって実現される。装置は、チップおよびオフチップ・メモリを備え、画像の圧縮または再構成を実行する。オフチップ・メモリのサイズを最小にして、クライアントサーバーのアプリケーションおよびホストープリンタのアプリケーションのようなアプリケーションにおける、圧縮および再構成のコストを削減する。また、チップおよびオフチップ・メモリ間の入出力トランザクション数も削減され、画像の圧縮および再構成を実行する速度が増加する。

【0021】図1および図2は、画像10から埋め込みビットストリーム（たとえば、画像のビットマップ）を生成する方法を示す。画像10はK本の行（ライン）を含み、ここでKは正の整数である。それぞれのレベルにおいて、サブバンド分解が一度にN行に対して実行される。ここでNは正の整数であり、 $1 < N < K$ 、かつNはKから独立している。図1は、N行の第1のウィンドウ12（実線で示す）およびN行の第2のウィンドウ14（点線で示す）を示す。これらのウィンドウおよび他のウィンドウは、画像10を上から下に、画像のK本すべての行に及ぶまで「タイルのように並べて」表示される。

【0022】また、図1は、第1のウィンドウ12および第2のウィンドウ14の重なりを示す。第1のウィンドウ12の少なくとも1つの行が、第2のウィンドウ14に保持される。これは、他のすべてのウィンドウについても同様である。処理されるそれぞれのウィンドウは、直前のウィンドウの少なくとも1行を含む。

【0023】ウィンドウは、連続して処理される（ブロック100）。こうして、第1のウィンドウ12が処理され、次に第2のウィンドウ14が処理され、次に第3のウィンドウ（図示せず）が処理される。このように、画像のK本の行すべてが分解されるまで、第1レベルにおける処理が続行する。画像の最上部から始まるウィンドウが、第1のウィンドウ12の前に与えられて処理されることは、理解されるであろう。

【0024】それぞれのウィンドウを、利用可能になると直ちに処理することができる。すべてのウィンドウが利用可能になるのを待ってから処理を開始する必要はない。

【0025】N行のそれぞれのウィンドウは、サブバンド分解を実行することにより処理される（ブロック10

2)。ウェーブレット変換を使用することができる。垂直高域および低域フィルタリングがウィンドウに対して実行され、次に水平高域および低域フィルタリングが実行される。結果として、高域水平、高域垂直サブバンドHH、高域水平、低域垂直サブバンドHL、低域水平、高域垂直サブバンドLH、および低域水平、低域垂直サブバンドLLについて、変換係数の1または複数の行16が得られる(図3を参照)。

【0026】十分な数の変換係数の行が利用可能になるまで、連続するウィンドウが分解される(ブロック104)。十分な数の変換係数の行が利用可能になると、3つのサブバンド(HH、HLおよびLH)の変換係数が量子化され(ブロック106)、順序付けられて(ブロック108)、エントロピー符号化される(ブロック110)。エントロピー符号化されたビットは、直ちにまたは後で、ビットストリームに配置される。

【0027】十分な数の変換係数の行が利用可能になると、最低サブバンド(LL)がさらに分解される(ブロック112)。この第2レベルの分解に続いて、最低サブバンドがさらに分解され、これより高い(すなわち高域)サブバンドが量子化され、順序付けられ、エントロピー符号化される。エントロピー符号化されたビットは、ビットストリームに配置される。

【0028】この方法を、チップおよびオフチップ・メモリを備えるハードウェアによって実現することができる。このように具現化されると、この方法により、オフチップ・メモリのサイズを低減することができる。画像全体を格納して処理するのとは異なり、一度に1つのウィンドウを格納して処理することができる。また、この方法は、チップおよびオフチップ・メモリ間の入出力トランザクション数を低減することもできる。

【0029】図4は、チップ22およびオフチップ・メモリ24を備え、画像10から埋め込みビットストリームを生成するシステム20を示す。チップ22およびオフチップ・メモリ24は、コンピュータのような第1のマシン26の一部であってもよい。

【0030】さらに、システム20は、スキヤナのような撮像装置28を備える。撮像装置28は、一度に複数行の画像データ(たとえば、画像の輝度またはクロミナンスデータ)を供給する。撮像装置28は、一度にN行を供給することができる。N行は、オフチップ・メモリ24にバッファされる。サーチューラバッファリング技法を使用して、それぞれのウィンドウが、直前のN行ウィンドウからの少なくとも1行を保持するようにすることができます。それぞれのN行ウィンドウは、オフチップ・メモリ24において利用可能になると、処理される。

【0031】チップ22は、サブバンド分解フィルタの第1ステージ30を備える。チップ22は、オフチップ・メモリ24からN行ウィンドウを読み取り、第1ステージ30は、複数のサブバンドHH、HL、LHおよび

LLを生成し、こうして、それぞれのサブバンドが、一度にS行生成される。ここで、Sは正の整数であり、S<Nである。たとえば、N=10かつS=1の場合、10行ウィンドウの分解により、それぞれのサブバンドHH、HL、LHおよびLLについて1行の変換係数が生成される。

【0032】さらに、チップ22は、高い方のサブバンドHH、HLおよびLHに対応する、第1、第2および第3の符号器32、34および36を備える。それぞれの符号器32、34および36は、対応するサブバンドのP行が利用可能になると直ちに、そのP行に対してエントロピー符号化を実行する(Pは正の整数であり、P>Sである)。このように、第1の符号器32は、サブバンドHHのP行が利用可能になったときにサブバンドHHを符号化し、第2の符号器34は、サブバンドHLのP行が利用可能になったときにサブバンドHLを符号化し、また第3の符号器36は、サブバンドLHのP行が利用可能になったときにサブバンドLHを符号化する。

【0033】さらに、チップ22は、サブバンド分解フィルタの、M個のさらなる継続ステージを備える。それぞれの付加ステージ30は、直前のステージで生成された最低サブバンドのN行ウィンドウに対して分解を実行する。第1、第2および第3の符号器32、34および36が、それぞれの付加ステージ30について設けられる。M番目のステージ30の最低サブバンドは、第4の符号器38によって符号化される。

【0034】それぞれの符号器32、34、36および38は、エントロピー符号化を実行する。また、それぞれの符号器32、34、36および38は、符号化の前に、量子化およびビットの順序付けを実行する。

【0035】さらに、チップ22は、SRAMのようなオンチップ・メモリ40を備える。オンチップ・メモリ40を、それぞれのステージ30によって実行されるフィルタリング操作に使用することができる。

【0036】また、チップ22は、コントローラ42を備え、オフチップ・メモリ24における画像の行のバッファリングを制御し、N行のウィンドウがバッファされたときを判断し、オフチップ・メモリ24からN行ウィンドウをアクセスする。代わりに、そのような制御を、撮像装置28によって、または第1のマシン26のホスト・プロセッサによって実行することもできる。

【0037】さらに、システム20は、復号器を有する第2のマシン44を備え、埋め込みビットストリームから画像を再構成する。埋め込みビットストリームの再構成については、図14および図15を参照して後で説明する。第2のマシン44がプリンタである場合、再構成された画像を印刷することができる。第2のマシン44がコンピュータである場合、再構成された画像をビデオモニタ上に表示することができる。

【0038】例として説明する操作では、大型画像が撮像装置28により走査され、1行ずつオフチップ・メモリ24に格納される。チップ22は、画像を埋め込みビットストリームに圧縮する。埋め込みビットストリームを、第1のマシン26に格納することができ、または直接第2のマシン44に送信して表示または印刷することもできる。復号器46は、埋め込みビットストリームから画像を再構成し、再構成された画像は、一度に複数行が表示または印刷される。

【0039】印刷操作中に埋め込みビットストリームが切り捨てられた場合、第2のマシン44によって少なくともいくつかのサブバンドが受信されているならば、第2のマシン44は、なお画像全体を表示または印刷することができる。しかし、切り捨てにより損失したサブバンドに含まれる詳細は、表示されない。

【0040】図5は、フィルタ・ステージ30の第1の実施形態をさらに詳しく示す。第1のステージ30は、垂直高域フィルタ50および垂直低域フィルタ52を備え、オフチップ・メモリ24にバッファされるそれぞれのN行ウインドウに対して、垂直高域および低域フィルタリングを実行する。垂直フィルタリングの結果は、全体の行が水平フィルタリングに利用可能になるまで、一対の遅延素子54および56によって一時的に遅延される。さらに、フィルタ・ステージ30は、第1の対の水平高域および低域フィルタ58および60を備え、垂直高域フィルタ50の出力に対して水平高域および低域フィルタリングを実行する。さらにフィルタ・ステージ30は、第2の対の水平高域および低域フィルタ62および64を備え、垂直低域フィルタ52の出力に対して水平高域および低域フィルタリングを実行する。

【0041】水平高域および低域フィルタ58、60、62および64は、従来のものである。これらは、全体の行を水平方向にフィルタリングすることができる。垂直フィルタ50および52は、畳み込みまたはリフティング(lifting)を実行することができる。Daubec h i e sの9/7および5/7の双直交フィルタのようなフィルタを、垂直フィルタリングに使用することができる。たとえば、Daubec h i e sの9/7双直交フィルタでは、9個の係数すべてが一方の垂直フィルタによって畳み込みに使用され、7個の係数すべてが他方の垂直フィルタによって畳み込みに使用される。係数ビットは、オフチップ・メモリ24に格納され、そこからアクセスされる。また、符号(sign)、有効性および現ビットプレーンに関する情報が、オンチップ・メモリ40に格納される。

【0042】ウインドウ・サイズ(すなわちN)は、フィルタ長の約2倍である。たとえば、5/3双直交フィルタでは、ウインドウ・サイズはN=10行であることができる。

【0043】水平フィルタリングを実行する前に、垂直

フィルタリングを実行することが好ましい。こうすることにより、垂直フィルタ50および52が、水平フィルタ58、60、62および64によってフィルタリングされる行より低いビット深さを持つ行に対してフィルタリングすることが可能になる。たとえば、垂直フィルタ50および52は、8ビットの第1のビット深さを有する行をフィルタリングし、水平フィルタ58、60、62および64は、16ビットの第2のビット深さを有する行をフィルタリングすることができる。

【0044】また、図5は、オフチップ・メモリ24およびフィルタ・ステージ30に対応する符号器32、34および36をも示す。チップは、行nから行(n+N-1)までをオフチップ・メモリ24から読み取り、N行ウインドウは、フィルタ・ステージ30によって分解される。フィルタ・ステージ30により生成されたサブバンド行は、オフチップ・メモリ24に書き込まれる。サブバンドの符号化のために十分な数の行が利用可能になると、そのサブバンドに対応する行がオフチップ・メモリ24から読み取られ、符号化される。

【0045】図6は、フィルタ・ステージ30aの代替の実施形態を示す。フィルタ・ステージ30aは、長さが5である5/3双直交フィルタを使用し、3行(n、n+1、n+2)のサイズを持つウインドウをフィルタリングする。フィルタ・ステージ30aは、垂直高域フィルタ50aを備え、3行ウインドウに対して垂直リフティング・ステップを実行する。垂直高域フィルタ50aの出力は、第1の遅延素子54aにより遅延され、遅延された出力は、一対の水平高域および低域フィルタ58aおよび60aによってフィルタリングされる。3行ウインドウの第1の行(n)が第2の遅延素子56aにより遅延され、第2の対の水平高域および低域フィルタ62aおよび64aによってフィルタリングされる。

【0046】さらに、フィルタ・ステージ30aは、垂直低域フィルタ52aを備え、第1および第2の対の水平高域および低域フィルタ58a、60a、62aおよび64aの出力に対して水平リフティングを実行する。最低サブバンドが、次のフィルタ・ステージに送られ、これより高いサブバンドは、符号器32、34および36に送られる。

【0047】このように、垂直高域および低域サンプルが、以下に示すように、連続したリフティング・ステップで形成される。

【0048】 $H_n(i) = 0.5[x_{2n+1}(i) - 0.5(x_{2n}(i) + x_{2n+2}(i))]$  および、 $L_n(i) = x_{2n}(i) + 0.5[H_{n-1}(i) + H_n(i)]$

ここで、 $x_n(i)$ は、画像のn番目の行のi番目のサンプルを表し、また、 $H_n(i)$ および $L_n(i)$ は、垂直高域および垂直低域サブバンドのそれぞれのi番目のサンプルを表す。これらの垂直サブバンドの行のそれぞれに対して、水平変換が通常の方法で適用される。フィルタ

・ステージ30aは、垂直方向にこれらのリフティング  
・ステップを実施する。水平変換は、垂直変換の第1および第2のリフティング・ステップの間で適用されるが、水平および垂直のフィルタリング操作は相互に交換することができるので、結果に影響することはない。水平変換を任意の方法で適用することができるが、リフティング・ステップにより、1バイト精度を保持することが可能となる。

【0049】サイズが3のウィンドウを処理することにより、および長さが5のフィルタを持つことにより、フィルタ・ステージ30aは、何らかのメモリ节约を得るために、メモリのニーズを、あるプレーンから他のプレーンに押し進める。このステージ30aは、本質的に、垂直フィルタリングを2つのステージで実行する。

【0050】垂直5/3双直交ウェーブレット変換の実施における第1のリフティング・ステップ(すなわちリフティング・ステップH<sub>n</sub>)は、画像の奇数行のサンプルを、これらのサンプルと、画像の隣接する偶数行上の真上および真下のサンプルの平均を取って得られた予測との差で置換することと解釈することができる。これらの予測残差が、実際には垂直高域サンプルである。圧縮性能は、高域帯域の信号エネルギーを低減できる程度に強く関連するので、さらに精密な予測を使用することにより、よい結果を得ることができる。特に、第1のリフティング・ステップを、次の式で置き換えることができる。

【0051】 $H_n(i) = 0.5[x_{2n+1}(i) - P(x_{2n}, x_{2n+2}(i))]$

ここで、演算子Pは、画像のローカル方向特性を利用して、隣接する2つの行 $x_{2n}(i)$ および $x_{2n+2}(i)$ の近傍サンプルから、奇数行サンプル $x_{2n+1}(i)$ を予測する。

【0052】垂直フィルタにはいくつかの異なる方向がある。それぞれの方向について、より小さい目標値を持つ方向が、基礎となるシーン(scene)を反映しやすくなるように、予測子(predictor)P<sup>i</sup>および目標値O<sup>i</sup>を計算する。追加のオンチップ論理回路を使用して、方向を選択することができる。以下のように、5つの方向がある。

【0053】垂直方向:

$$P^0(i) = 0.5[x_{2n}(i) + x_{2n+2}(i)]$$

$$O^0(i) = \sum \|x_{2n}(i+k) - x_{2n+2}(i+k)\|, k = -2 \sim k$$

左上から右下に-45°:

$$P^{-45}(i) = 0.5[x_{2n}(i-1) + x_{2n+2}(i+1)]$$

$$O^{-45}(i) = \sum \|x_{2n}(i+k-1) - x_{2n+2}(i+k+1)\|, k = -2 \sim k$$

左上から右下に+45°:

$$P^{+45}(i) = 0.5[x_{2n}(i+1) + x_{2n+2}(i-1)]$$

-1)]

$$O^{+45}(i) = \sum \|x_{2n}(i+k+1) - x_{2n+2}(i+k-1)\|, k = -2 \sim k$$

左上から右下に-23°:

$$P^{-23}(i) = 0.5[x_{2n}(i-0.5) + x_{2n+2}(i+0.5)]$$

$$O^{-23}(i) = \sum \|x_{2n}(i+k-0.5) - x_{2n+2}(i+k+0.5)\|, k = -2 \sim k$$

左上から右下に+23°:

$$P^{+23}(i) = 0.5[x_{2n}(i+0.5) + x_{2n+2}(i-0.5)]$$

$$O^{+23}(i) = \sum \|x_{2n}(i+k+0.5) - x_{2n+2}(i+k-0.5)\|, k = -2 \sim k$$

$O^{m i n}(i)$ がこれらの5つの目標値の最小値を表し、 $P^{m i n}(i)$ がそれぞれのサンプル位置における対応する予測子を表すとする。単純でロバストな方向適応予測子を、次の式に従って通常の垂直補間を $P^{m i n}$ の値と組み合わせることにより形成することができる。

【0054】

$$P(i) = \alpha(i)P^0(i) + (1-\alpha)P^{m i n}(i)$$

ここで、混合パラメータ $\alpha(i)$ は、次の式により与えられる。

【0055】

$$\alpha(i) = [O^{m i n}(i) + 1] / [O^0(i) + 1]$$

混合パラメータ $\alpha(i)$ を、次のように変更することができる。

$$\alpha(i) = \max\{1, [O^{m i n}(i) + 2 \min\{0, O^0(i) - O'(i)\} + 2 \min\{0, O^0(i) - O''(i)\} + 1] / [O^0(i) + 1]\}$$

【0057】ここで、 $O'(i)$ および $O''(i)$ は、最適角度が正の場合は負の角度目標値 $O^{-45}$ および $O^{-23}$ により特定され、最適角度が負の場合は正の角度目標値 $O^{+45}$ および $O^{+23}$ により特定され、その他の場合は垂直目標値 $O^0$ により特定される。変更された混合パラメータは、基礎となるサンプル値の連続的な適度に滑らかな関数であり、これにより、復号器における量子化の影響に対する感受性が小さくなる。このことは、ローカル方向が不明瞭なときに、垂直方向以外のものの影響を抑制する効果を有する。

【0058】次に、図7および図8を参照する。これらの図は、それぞれの符号器32、34、36および38によって使用されるブロック符号化技法を説明する。所与のサブバンド(たとえば、HHサブバンド)のブロック符号化において、チップ22は、変換係数の行をオフチップ・メモリ24に書き込む(ブロック202)。所与のサブバンドのP個の行が利用可能になると(ブロック204)、符号器32が、オフチップ・メモリ24からP行をアクセスし(ブロック206)、HHサブバンドのP行を複数のブロックとして符号化する(ブロック208)。それぞれのP行から生成されたブロックは、

相互に独立して符号化される。一般的なブロック・サイズは、 $64 \times 64$ サンプル、 $32 \times 32$ サンプル、 $64 \times 16$ サンプルまたは $128 \times 8$ サンプルである。

【0059】埋め込みビットストリームは、1998年12月17日出願の米国特許出願第09/213,743号の「DECODING OF EMBEDDED BITSTREAMS PRODUCED BY CONTEXT-BASED ORDERING AND CODING OF TRANSFORM COEFFICIENT BIT-PLANES」（「Ordentlichによる特許出願」）に従い、サブバンド分解ブロックを別個に符号化することにより生成される。Ordentlichによる特許出願を、ここで参照により取り入れる。それぞれのブロックの符号化は、変換係数を量子化するステップと、量子化された係数のビットプレーンを生成するステップと、ビットプレーンを異なるサブシーケンス（たとえば、非ゼロの近傍(Non-Zero Neighbors)、非ゼロの親(Non-zero Parents)、ラン(RUN)およびリファインメント(Refinement)）に分解するステップと、記述の期待されるビットあたりの期待される歪み減少量が減っていくように、サブシーケンスを順序付けるステップと、順序付けられたサブシーケンスを符号化するステップと、サブシーケンスを順序どおりにビットストリームに配置するステップとを含むステップにより実行される。また、ブロックの所与のビットプレーンを順序付けるコンテキストを、該ブロックの、前に符号化されたビットプレーン内に完全に含めることができる。また、ブロックの所与のビットプレーンを順序付けるコンテキストを、「親ブロック」の、前に符号化されたビットプレーン内に含めることもできる。用語「親ブロック」とは、より低い解像度のサブバンドの対応する空間的方向にあるブロックを意味する。

【0060】図9は、変換係数のそれぞれのブロックについて埋め込みビットストリームを生成する、より詳細な方式を示す。それぞれのブロックは、4分木(quad tree)分解およびビットプレーン符号化の組み合わせにより符号化される。所与のブロックに対して4分木分解を実行し、所与のブロックが複数のサブブロックに分解されるようにするステップ（ブロック302）と、有効係数を含むサブブロックを識別するステップ（ブロック304）と、有効係数を含むすべてのサブブロックをビットプレーン符号化するステップ（ブロック306）と、有効でない係数を持つブロックに対して追加の4分木分解を実行するステップ（ブロック308）とを含むステップによって、所与のブロックは符号化される。

【0061】次に、他のレベルの4分木分解およびビットプレーン符号化が、有効係数を持たないそれぞれのサブブロックに対して実行される。このように、有効係数を持たないサブブロックは、より小さなサブブロックにさらに分解される（ブロック310）。

【0062】有効係数を持つこれらのより小さなサブブロックは、ビットプレーン符号化され（ブロック30

6）、有効係数を持たないより小さなサブブロックは、さらにもっと小さなブロックに分解される（ブロック308）。

【0063】典型的なサブブロックが、4分木分解によって生成される最小サブブロックであるサイズ $G \times G$ まで分解されると（ブロック312）、 $G \times G$ サブブロックおよび任意のこれより小さなサブブロックが、ビットプレーン符号化される（ブロック314）。画像の境界上に存在するサブブロックの中には、 $G \times G$ より小さいものがある。たとえば、境界上のサブブロックには、 $G \times 2$ 、 $G \times 1$ 、 $1 \times G$ などのサブブロックが存在することがある。

【0064】サブブロックを、Ordentlich等の特許出願に記載されているブロック符号化技法を使用して、ビットプレーン符号化することができる。

【0065】このハイブリッドな符号化方式の例を、図10に示す。図10には、ブロック400に対する4分木分解の2つのレベルが示されている。最初に、ブロック400は、 $64 \times 64$ サンプルを含む。4分木分解の第1のレベルが、 $32 \times 32$ のサンプル・サブブロックを4つ、すなわちA、B、CおよびDを形成する。サブブロックBおよびCは、有効係数を持つことが判明したので、ビットプレーン符号化される。次に、サブブロックAが、より小さな $16 \times 16$ のサンプル・サブブロックA11、A12、A21およびA22に分解され、サブブロックDが、より小さな $16 \times 16$ のサンプル・サブブロックD11、D12、D21およびD22に分解される。第2のレベルの4分木分解の後、サブブロックA11、A21およびD11が重要な係数を持つことが判明したので、ビットプレーン符号化される。残りのサブブロックA12、A22、D12、D21およびD22は、さらにより小さな $8 \times 8$ に分解される。

【0066】ブロック符号化を実行し、埋め込みビットストリームを生成するさらに詳しい方式については、参考としてここで取り入れられる、この発明と同時出願された米国特許出願第09/267 248号の「EMBEDDED BLOCK CODING WITH OPTIMIZED TRUNCATION」（代理人整理番号10990121-1）で開示される。

【0067】ブロック符号化は、画像を小さなタイルに分割してそれぞれのタイルを独立的に圧縮する操作と同じものではない。符号化は、ブロックに対して独立的に実行されるが、変換は、ブロックごとに実行されない。この理由は、ブロックのアーティファクトの発生を防ぐためである。ブロック符号化および独立したタイル圧縮の間の他の相違点は、異なるサブバンドで符号化されたブロックは、通常、元の画像の同じ領域を表さないということである。すべての帯域でブロック・サイズが同一の場合には、低い解像度帯域にあるブロックが、元の画像の大きな領域を連続的に表す。

【0068】符号化において、3つのサブバンドLH、

HLおよびHHのそれれにおけるブロックの位置を、任意の所与の解像度レベルで互い違い（交互配置）にすることができる。サブバンドHL、LHおよびHHの垂直境界を相互にP/3行だけオフセットすることにより、ブロックの交互配置を実行することができる。たとえば、サブバンドHLをP/3行だけオフセットし、サブバンドLHを2P/3行だけオフセットする。サブバンドHHのP行がオフチップ・メモリ24に格納されているとき（したがって符号化に利用可能である）、サブバンドHLの2P/3行、およびサブバンドLHのP/3行だけがオフチップ・メモリ24に格納される。サブバンドHLのP行が符号化に利用可能になったときは、サブバンドLHの2P/3行、およびサブバンドHHのP/3行だけがオフチップ・メモリ24に格納されている。サブバンドLHのP行が符号化に利用可能になったときは、サブバンドHHの2P/3行、およびサブバンドHLのP/3行だけがオフチップ・メモリ24に格納されている。

【0069】図11は、サブバンドHL、LHおよびHHが相互にP/3行ずつ交互配置される方法を示す。カウンタ450、452および454を、所与のフィルタ・ステージの3つのサブバンドHH、HLおよびLHについて設けることができる。カウンタ450、452および454は、異なる値0、P/3、2P/3で初期化される。所与のサブバンドについて行が outputされるたびに、所与のサブバンドに対応するカウンタが増分される。カウンタがブロック・サイズPより大きくなると、そのカウンタはゼロに設定され、所与のサブバンドからのブロックが符号化される。たとえば、3つのサブバンドのうち2つについて第1のストライプ(stripe)符号化を実行中、対応するカウンタがPに等しい場合でも、符号化に使用可能な行数はPより少ない。このように、第1のストライプ符号化中に、1つのサブバンドのP行が符号化され、もう1つのサブバンドの2P/3行が符号化され、また第3のサブバンドのP/3行が符号化される。

【0070】ブロックの交互配置により、符号化に使用されるオフチップ・メモリ24の合計量がP行だけ削減される。また、ブロック符号化は、ブロック符号化の負荷が時間にわたって均一に分散されるので、符号化の「ジッター」を減少させる。この結果、一定のレートで画像の行を生成または使用するアプリケーションの「ジッター」を吸収するのに割り当てる追加のメモリ量が削減される。

【0071】ブロック符号化には潜在的な利点が多数含まれる。たとえば、ブロックを選択的に伝送して、関心のある領域(regions-of-interest)を再構成することができる。

【0072】次に、関心のある領域の再構成を実行するシステムおよび方法を示す図12および図13を参照す

る。図12は、ネットワーク506上で通信するサーバー502およびクライアント504を備えるシステム500を示す。ネットワーク506は、ローカル・エリア・ネットワークからインターネットまで任意のものでよい。

【0073】サーバー502は、プロセッサ508およびメモリ510を備える。サーバーのメモリ510は、画像のサブバンド分解512で符号化される。サブバンド分解512は、最低解像度サブバンドと、これより高い解像度の複数のサブバンドを含む。それぞれのサブバンドは、複数のブロックとして符号化され、それぞれのブロックは、画像領域を表す。領域の異なる解像度は、異なるサブバンド・ブロックに表される。

【0074】サーバーのメモリ510は、プログラム514でさらに符号化される。このプログラム514は、画像に対するネットワーク要求に応答して、最低解像度サブバンドをネットワーク506上に出すよう、サーバー・プロセッサ508に命令する。さらに、サーバー・プログラム514は、画像領域に対するネットワーク要求に応答して、少なくとも1つの追加のブロックをネットワーク506上に出すよう、サーバー・プロセッサ508に命令する。追加の1または複数のブロックは、要求された領域についてさらに高い解像度を提供する。

【0075】クライアント504は、第2のプロセッサ516、およびプログラム520を格納するためのメモリ518を備える。クライアント・プログラム520は、画像および画像領域に対するネットワーク要求を送信するよう、クライアント・プロセッサ516に命令する。ネットワーク要求を、ユーザーによって対話的に生成することができる。

【0076】さらに、クライアント・プログラム520は、ネットワーク要求に応答してサーバー502が送信したブロックを受信し、ネットワーク506上で受信したブロックから画像および関心のある領域を再構成するよう、クライアント・プロセッサ516に命令する。

【0077】図12は、ソフトウェアを介して画像を完全に再構成するクライアント504を示す。代替方法として、クライアント504は、チップおよびオフチップ・メモリを備えて、画像および関心のある領域を再構成することができる。

【0078】図13は、サーバー502およびクライアント504の間のトランザクション例を示し、この例では、クライアント504が、国のある地域に関する情報を要求する。クライアント504上で走るのは、ウェブ・ブラウザのようなプログラムである。ユーザーは、国の地図のURLを入力し、クライアント504は、地図の要求をネットワーク506上に出す（ブロック602）。

【0079】地図のサブバンド分解を格納するサーバー502は、ネットワーク要求を受信し、分解の低サブバ

ンドにおけるブロックをアクセスし、すべての低サブバンド・ブロックをクライアント504に送信する（ブロック604）。

【0080】クライアント504はブロックを受信し、地図全体の低解像度画像を再構成する（ブロック606）。低解像度の地図がユーザーに表示される。マウスのような入力装置を使用して、ユーザーは地図のある領域をクリックし、クライアントは、関心のある領域に対する要求を生成する。要求はサーバー502に送信される（ブロック608）。

【0081】サーバー502は要求を受信し、関心のある領域に対応するブロックをアクセスする。ブロックは、異なるサブバンドから取り出される。サーバー502は、より高いこれらのサブバンド・ブロックをクライアント504に送信する（ブロック610）。

【0082】クライアント504はブロックを受信し、関心のある領域を再構成する（ブロック612）。関心のある領域がユーザーに表示される。再構成される領域の解像度（すなわち、関心のある領域の詳細）は、サーバー502によってアクセスされるサブバンドに依存する。

【0083】このクライアントサーバーのアプリケーションは、リモートに位置するサーバー上に常駐して、画像のより小さな領域に関心を持つ個々のクライアントによって対話的にアクセスされ再構成される大型圧縮画像を含む。サーバー502は、関心のある領域に関連するより高い解像度のコードブロックを送信するだけでよい。クライアントに送信されるブロックによってカバーされるサブバンド・サンプルのセットを判断する際には、サブバンド合成フィルタがサポートする領域を考慮に入れるべきである。

【0084】低解像度レベルにおけるブロックは、画像の本質的な部分に広がり、クライアントが実際に要求したもの以上の情報が伝送されるようにする。これは、関心のある領域が正しく再構成されることを確実にするよう実行される。しかし、ほとんどの関心のある領域の復号化のアプリケーションでは、対話ユーザーは、画像のより大きな領域を横切って移動することができるので、前に受信されたコード・ブロックが適切にキャッシュされて再利用されるならば、このような操作中に伝送される新しい情報量は、最終的にカバーされるその領域のサイズとほぼ同じ大きさになる。

【0085】それぞれのサブバンドのそれぞれのブロックを連続して伝送することができ、利用可能なネットワーク帯域幅を、関心のある領域の画質が徐々に改善されるよう使用することができる。このことは、対話型ユーザーが、新しい関心領域に移動する前にその領域に費やす時間の長さに依存する。これは、大部分の消費者がきわめて低い帯域幅のリンクしか持っておらず、またあまり辛抱強くない場合に、インターネット上で大型画像を

対話的に走査検索するには、特に有効な特徴である。

【0086】ブロック符号化の他のクライアントサーバーのアプリケーションは、大型画像の領域を、クライアント要求の累積履歴に基づき選択的に改善（リファイン； refine）する操作を含む。画像全体または画像の大きな部分が要求される場合には、対話型ユーザーがそれらの領域に主に関心を示すであろうという期待により、過去に最も頻繁に要求されたそれらの画像領域に対応するブロックに、より多くの割合を割り当てることができる。こうして、関心のある領域は、クライアントの好みの予測を見越して、より高い画質レベルで伝送される。さらに、サーバーの記憶スペースが密になって、さらにデータを収容しなければならない場合、あまり頻繁に要求されなかつた画像のブロックに対応するビットストリームを切り捨てて、スペースを開放することができる。

【0087】次に、埋め込みビットストリームから画像10を再構成する方法を示す図14を参照する。埋め込みビットストリームを生成するのに使用される方法を逆に実行することにより、画像を再構成することができる。たとえば、ビットストリームが、図1のフィルタリング方法および図7のブロック符号化方法（ここでは、Ordentlich等の特許出願に記載された方法に従い、ブロックは量子化され、順序付けられ、符号化される）を介して生成された場合、画像は以下の手順によって再構成される。すなわち、ビットストリームを復号化するステップ（ブロック702）と、復号化されたビットストリームにおけるサブシーケンスのビットを判断するステップと（ブロック704）、判断されたサブシーケンスからビットプレーンを満たすステップ（ステップ706）とによって再構成される。ステップ706において、ブロックの所与のビットプレーンを満たすコンテキストを、該ブロックの、前に復号化されたビットプレーン内に完全に含めることができ、また親ブロックの、前に復号化されたビットプレーン内に含めることもできる。

【0088】次に、量子化変換係数がビットプレーンから再構成される（ブロック708）。再構成された変換係数の値は、利用可能な不確定間隔の中間値であってもよい。

【0089】ブロック702からブロック708によって示されるステップが実行されるとき、再構成された変換係数は、逆変換フィルタリングに利用可能になる。画像全体が再構成される場合、再構成された変換係数は、行単位で利用可能になっていく。こうして、所与の分解レベルのそれぞれのサブバンドについての変換係数の第1の行が、図3に示すように現れることができる。関心のある領域が再構成される場合は、関心のある領域に対応する変換係数ブロックだけが利用可能になる。

【0090】十分な数の、部分的または完全な行が利用可能になると（すなわち、N行のチャunk（かたまり）が形成されると）、逆水平フィルタリングが、それぞれ

のN行チャンクに対して実行され（ブロック710）、また逆垂直フィルタリングが、水平方向にフィルタリングされたチャンクに対して実行される（ブロック712）。

【0091】画像全体または関心のある領域は、いくつかのパス（工程）で再構成される。それぞれのパスの中で、新しい情報が一度に複数行表示される。このように、低解像度の画像または関心のある領域が、第1のパスにおいて一度に複数行再構成される。第2およびその後のパスにおいて、詳細が一度に複数行追加される。埋め込みビットストリームが切り捨てられた場合、すでに受信されているすべての完全ブロックが表示される（これは、ブロックが、相互に独立して符号化されるからである）。こうして、ビットストリームが切り捨てられた場合、画像または関心のある領域の上半分は比較的鮮明な特徴を含み、下半分は不鮮明な特徴を含むことがある。

【0092】次に図15を参照すると、画像再構成を装置800によって実行することができ、装置800は、オフチップ・メモリ802およびチップ804を備え、チップ804は、複数の逆フィルタリング・ステージ806と、それぞれのフィルタリング・ステージについての復号器808、810および812を備える。第1のフィルタリング・ステージは、最低サブバンドL1について追加の復号器814を備える。オンチップ・メモリ816は、逆フィルタリング操作に使用される。オンチップ・コントローラ818は、変換係数のチャンクが逆フィルタリングに利用可能になる時を判断することができる。チップ804を、クライアントのビデオ・カードに搭載することができる。代替の実施形態では、画像再構成を、ホスト・プロセッサ（たとえばクライアント・プロセッサ516）により、すべてソフトウェアで実行することができる。再構成された画像を、表示または印刷の前に、外部メモリ（たとえば、オフチップDRAM、ハード・ドライブ、ビデオ・メモリ）にバッファすることができる。

【0093】ブロック交互配置が画像圧縮に使用された場合、画像の再構成中に、ブロック交互配置を逆に実行する必要はない。しかし、画像再構成中にブロック交互配置を実行すると、オフチップ・メモリの節約になる。変換係数サブバンドのオフチップ・メモリ802への書き込みを、相互にP/3行だけオフセットすることができる。オフチップ・メモリ802においてサブバンドのチャンクが利用可能になると直ちに、対応する復号器が、当該チャンクを量子化された変換係数に復号する。

【0094】4分木分解およびビットプレーン符号化の組み合わせが画像圧縮において使用された場合は、それぞれの受信されたサブブロックのビットプレーン符号化が逆に実行され、4分木復号化は、サブブロックをブロックに再度組み立てるよう逆に実行される。

【0095】画像圧縮中に垂直フィルタを回転した場合、画像の再構成中に垂直フィルタを回転する必要はない。しかし、垂直フィルタを回転させると、再構成された画像の画質が改良される。しかし、変更されたリフティング・ステップが逆に実行されるとき、関連する方向適応予測子が、偶数行サンプルの幾分歪んだコピーから導き出される点に注意すべきである。これは、画像圧縮が、量子化エラーを取り入れると予測されるからである。要するに、演算子Pを、その入力における小さなエラーを極力感知しないようにするべきである。

【0096】以上のように、画像全体を格納することなくサブバンド分解を実行する発明が開示される。結果として、この発明は、画像圧縮およびその後の再構成に使用されるオンチップ・メモリおよびオフチップ・メモリの両方のサイズを削減する。またこの発明は、チップおよびオフチップ・メモリ間の入出力トランザクション数をも削減する。オフチップ・メモリは、10行から数百行の間の行を格納することができ、任意のサイズの画像を処理する。入出力メモリ帯域幅は、画像あたり約3および5バイトのトランザクションに削減することができる。さらに、チップおよびオフチップ・メモリは、画像のサイズによる制限を受けずに、異なるアプリケーションに使用することができる。

【0097】また、チップは、埋め込みビットストリームを生成する。埋め込みビットストリームは、たとえば圧縮率が適切ではなく、そのためには画像の微細な詳細のみを切り捨てる場合に有利である。

【0098】リフティング・ステップおよび1行ずつの階層化されたエントロピー符号化の組み合せは、メモリの使用をさらに削減する。きわめて小さな垂直領域をサポートするウェーブレット変換を使用することにより、メモリ要件はさらに削減される。

【0099】異なるサブバンドについてのブロック・パートィションの交互配置は、ブロック符号化のアプリケーションにおいてメモリの使用を削減する。画像におけるウェーブレットの因子分解(factorization)およびローカル方向特性の利用は、メモリ要件を増やすことなく、優れた圧縮パフォーマンスの達成を可能にする。

【0100】また、ブロック符号化の使用は、オフチップ・メモリのサイズを増加することなく、かつチップおよびオフチップ・メモリ間の入出力トランザクション数を増加することなく、圧縮パフォーマンスを向上させる。さらに、ブロック符号化は、関心のある領域の分解のような特別機能も提供する。

【0101】この発明は、上記に説明したホスト・プリンタおよびクライアント・サーバーのアプリケーションに限定されるものではない。たとえば、この発明を、衛星画像に適用することができる。さらに、この発明を、新しいJPEG-2000標準と互換性のあるアプリケーションに適用することができる。

【0102】この発明は、上記に説明し図示された特定の実施形態に制限されるものではない。代わりに、この発明は、特許請求の範囲に従って解釈されるものである。

【0103】この発明は、以下の実施態様を含む。

【0104】(1) K行 (Kは正の整数) を含む画像を圧縮する装置であって、前記画像のN行 (Nは正の整数、 $1 < N < K$ 、NはKから独立している) ウィンドウを順次提供する手段と、垂直高域フィルタおよび垂直低域フィルタを有するサブバンド分解フィルタの第1のステージであって、前記提供されたそれぞれのN行ウィンドウに対して垂直高域および垂直低域フィルタリングを実行する第1のステージを有するチップとを備え、前記画像全体が複数のサブバンドに分解されるまで、前記チップが、前記画像のN行を一度にフィルタリングするようにした画像圧縮装置。

【0105】(2) 前記提供する手段が、前記ウィンドウをバッファする画像バッファを備え、それぞれのウィンドウが、前のウィンドウからの少なくとも1行を保持するようにした上記(1)に記載の画像圧縮装置。

【0106】(3) 前記Nが、フィルタ長の約2倍である上記(1)に記載の画像圧縮装置。

【0107】(4) 前記チップは、サブバンド分解フィルタのM個の付加継続ステージを備え、それぞれの付加ステージは、垂直高域フィルタおよび垂直低域フィルタを備えて、複数のサブバンドを生成し、前のステージによって生成された最低サブバンドのN行ウィンドウに対して垂直高域および垂直低域フィルタリングを実行するようにした上記(1)に記載の画像圧縮装置。

【0108】(5) 前記垂直フィルタは、第1のビット深さを持つ画像行を受信して、該第1のビット深さより大きい第2のビット深さを持つ行を出力し、前記第1のステージは、前記垂直高域フィルタの出力に応答する水平高域および水平低域フィルタの第1の対、および前記垂直低域フィルタの出力に応答する水平高域および水平低域フィルタの第2の対を備えるようにした上記(1)に記載の画像圧縮装置。

【0109】(6) 前記第1のステージは複数のサブバンドを生成し、それぞれのサブバンドは一度にS行 (Sは正の整数) 生成され、該サブバンドのうちの1つがさらに分解され、該サブバンドのうちの他のサブバンドは一度にP行 (Pは正の整数) 符号化され、 $S < N$  であり、 $P > S$  である上記(1)に記載の画像圧縮装置。

【0110】(7) 前記チップは複数のエントロピー符号器を備え、それぞれのエントロピー符号器は、前記他のサブバンドのうちの1つに対応し、対応するサブバンドのP行が利用可能になると直ちに該対応するサブバンドを符号化するようにした上記(6)に記載の画像圧縮装置。

【0111】(8) 前記それぞれの符号器が、サブバン

ドのP行を複数のブロックとして符号化し、ブロックは、相互に独立して符号化されるようにした上記(6)に記載の画像圧縮装置。

【0112】(9) それぞれのブロックは、4分木分解およびビットプレーン符号化の組み合わせによって符号化されるようにした上記(8)に記載の画像圧縮装置。

【0113】(10) 所与のブロックに対して4分木分解を実行して、該所与のブロックが複数のサブブロックに分解されるようにするステップと、有効係数を含むサブブロックをビットプレーン符号化するステップと、有効でない係数を含むサブブロックに対して追加の4分木分解を実行するステップとを含むステップにより、前記所与のブロックが符号化されるようにした上記(9)に記載の画像圧縮装置。

【0114】(11) 所与のブロックの変換係数を量子化するステップと、前記量子化された係数のビットプレーンを生成するステップと、前記ビットプレーンを異なるサブシーケンスに分解するステップと、記述の期待されるビットあたりの期待される歪み減少量が減っていくように、前記サブシーケンスを順序付けるステップと、前記順序付けられたサブシーケンスを符号化するステップと、前記サブシーケンスを順序通りビットストリームに配置するステップとを含むステップにより、前記所与のブロックが符号化されるようにした上記(8)に記載の画像圧縮装置。

【0115】(12) オフチップ・メモリを備え、前記第1のステージは、直ちに符号化するためのT個 (Tは正の整数および $T > 1$ ) のサブバンドを出力し、前記画像圧縮装置は、前記サブバンドの垂直境界を相互にP/T行だけオフセットする手段を備え、サブバンドのブロックが相互に交互配置されるようにした上記(8)に記載の画像圧縮装置。

【0116】(13) 画像のサブバンド分解を実行するチップであって、該画像はK行 (Kは正の整数) を含んでおり、該チップは、前記画像のN行ウィンドウ (Nは正の整数、 $1 < N < K$  であり、NはKから独立している) を順次受け取り、前記チップは、L個 (Lは整数で $L > 1$ ) のステージ・フィルタを備えており、前記L個のステージ・フィルタのそれぞれのステージが、前記チップによって受け取られたそれぞれのN行ウィンドウに対して垂直高域フィルタリングを実行する第1の手段と、前記チップによって受け取られたそれぞれのN行ウィンドウに対して垂直低域フィルタリングを実行する第2の手段と、前記第1の手段の出力に対して水平高域フィルタリングを実行する第3の手段と、前記第2の手段の出力に対して水平低域フィルタリングを実行する第4の手段とを備え、前記画像全体が異なるレベルの複数のサブバンドに分解されるまで、前記チップが、前記画像のN行を一度にフィルタリングするようにしたチップ。

【0117】(14) 画像のサブバンド分解を実行する

フィルタ・ステージであって、該画像はK行 (Kは正の整数) を含んでおり、前記画像のN行 (Nは正の整数であり、N<K) に対して垂直リフティング・ステップを実行する第1のフィルタと、N行の第1の行に応答する水平高域および水平低域フィルタの第1の対と、垂直高域フィルタの出力に応答する水平高域および水平低域フィルタの第2の対と、前記水平高域および水平低域フィルタの第1の対の出力に対して水平リフティングを実行する第2のフィルタと、低域サブバンドを提供する前記第1のステージの第1の出力と、高域サブバンドを提供する前記第1のステージの複数のさらなる出力とを備えるフィルタ・ステージ。

【0118】(15) 前記第1および第2のフィルタが、複数の特定方向に回転可能であるようにした上記(14)に記載のフィルタ・ステージ。

【0119】(16) 画像のサブバンド分解を実行する方法であって、該画像はK行 (Kは正の整数) を含んでおり、前記画像のN行ウィンドウ (Nは正の整数であり、1<N<<Kであり、NはKから独立している) を順次提供するステップと、提供されたそれぞれのN行ウィンドウに対して垂直高域フィルタリングを実行するステップと、提供されたそれぞれのN行ウィンドウに対して垂直低域フィルタリングを実行するステップと、前記画像全体が複数のサブバンドに分解されるまで、前記N行ウィンドウの提供および垂直フィルタリングを繰り返すステップとを含む、サブバンド分解を実行する方法。

【0120】(17) サブバンド分解から画像を再構成するチップであって、該画像はK行 (Kは正の整数) を含み、前記サブバンド分解はQレベル (Qは1より大きい正の整数) を含み、それぞれのレベルは複数のサブバンドを含んでおり、前記チップが、所与の分解レベルのそれぞれのサブバンドについて複数のN行チャンクを受信する入力と、Q個の逆フィルタ・ステージとを備え、それぞれの逆フィルタ・ステージは、N行のそれぞれのチャンクに対して操作可能な逆水平高域および逆水平低域フィルタと、逆垂直高域および逆垂直低域フィルタの第1および第2の対とを備え、該第1の対は、前記水平高域フィルタの出力に応答し、該第2の対は、前記水平低域フィルタの出力に応答し、該垂直フィルタの第1および第2の対は、一度に複数行再構成された画像を出力するようにしたチップ。

【0121】(18) 埋め込みビットストリームから画像を再構成する方法であって、該埋め込みビットストリームは、該画像の符号化されたサブバンド分解を表しており、該画像はK行 (Kは正の整数) を含み、該サブバンド分解は複数の分解レベルを含み、それぞれのレベルは複数のサブバンドを含んでおり、前記埋め込みビットストリームを複数行に復号化するステップと、所与の分解レベルの前記サブバンドのそれぞれについて、N行のチャンクを形成するステップと、前記N行のそれぞれの

チャンクに対して逆水平フィルタリングを実行するステップと、前記水平フィルタリングされたチャンクに対して逆垂直フィルタリングを実行するステップとを含む、所与の分解レベルに対するステップを備える画像を再構成する方法。

【0122】(19) 前記埋め込みビットストリームが複数のブロックに復号化され、ブロックが相互に独立して復号化されるようにした上記(18)に記載の方法。

【0123】(20) 前記復号化が、逆4分木分解およびビットプレーン復号化を含むようにした上記(18)に記載の方法。

【0124】

【発明の効果】オフチップ・メモリのサイズを削減することができる。また、チップおよびオフチップ・メモリ間の入出力操作数を削減することができる。

【図面の簡単な説明】

【図1】一度にN行のウィンドウが変換される画像を示す図。

【図2】この発明に従う、画像を記述する埋め込みビットストリームを生成する方法のフローチャート。

【図3】画像のサブバンド分解を示す図。

【図4】チップおよびオフチップ・メモリを備え、埋め込みビットストリームを生成する装置のブロック図。

【図5】装置の符号器およびオフチップ・メモリに対応する、フィルタ・ステージのブロック図。

【図6】装置の符号器およびオフチップ・メモリに対応する、代替のフィルタ・ステージのブロック図。

【図7】複数のサブバンド行を複数の独立したブロックとして符号化する方法のフローチャート。

【図8】複数の独立したブロックとして符号化された複数のサブバンド行を示す図。

【図9】複数のサブバンド行を複数の独立したブロックとして符号化する、4分木符号化およびビットプレーン符号化の組み合わせを含む代替方法のフローチャート。

【図10】ブロックの4分木分解を示す図。

【図11】符号化に先立つブロックの交互配置を示す図。

【図12】埋め込みビットストリームから画像を再構成するシステムのブロック図。

【図13】図12のシステムが実行するトランザクション例のフローチャート。

【図14】この発明に従って埋め込みビットストリームから画像を再構成する方法のフローチャート。

【図15】チップおよびオフチップ・メモリを備え、埋め込みビットストリームから画像を再構成する装置のブロック図。

【符号の説明】

10 画像

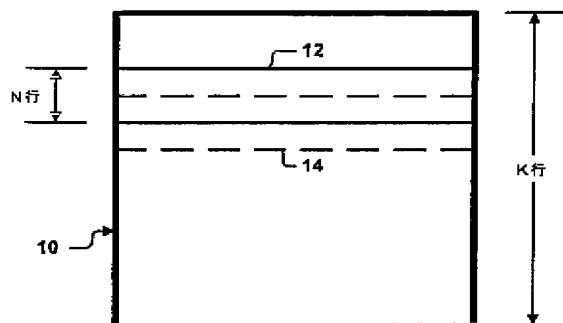
20、500 システム

22、804 チップ

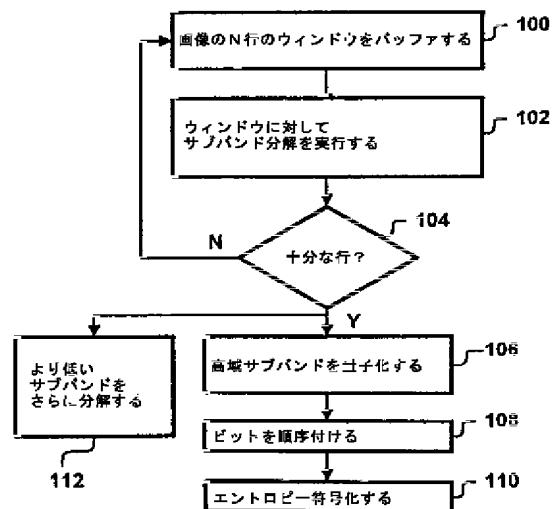
24、802 オフチップ・メモリ  
30 フィルタ・ステージ  
32 第1の符号器

34 第2の符号器  
36 第3の符号器  
38 第4の符号器

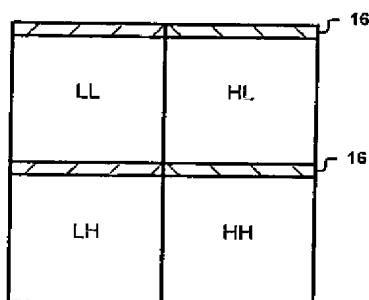
【図1】



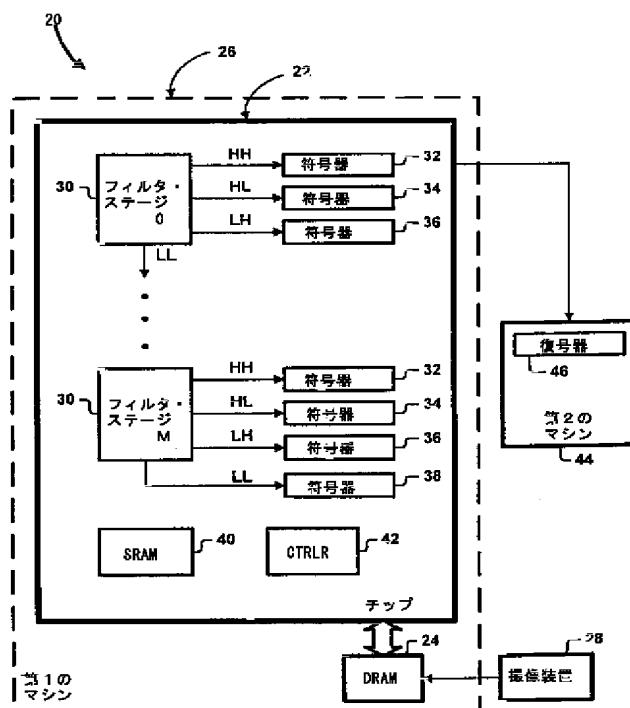
【図2】



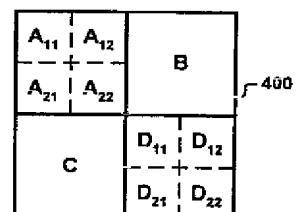
【図3】



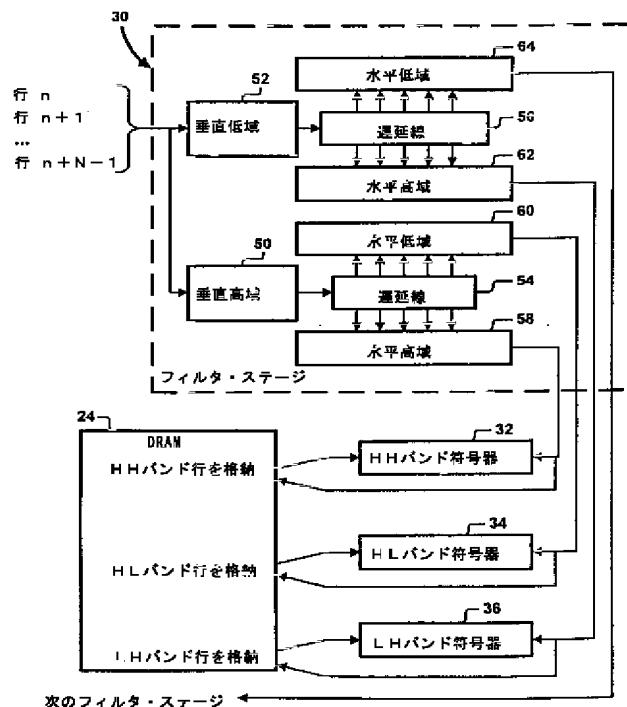
【図4】



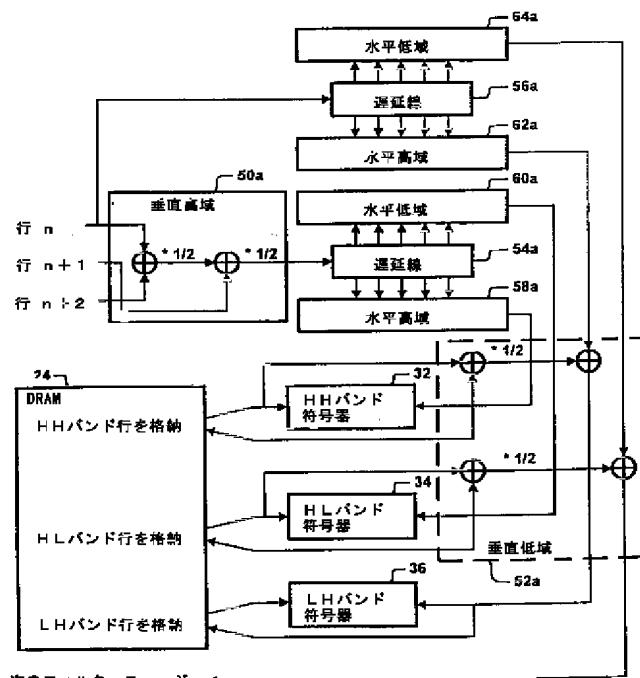
【図10】



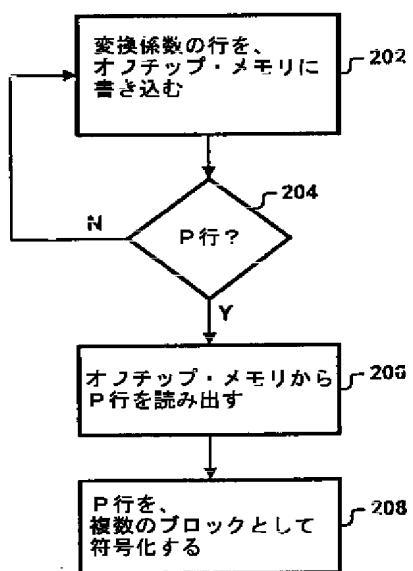
【図5】



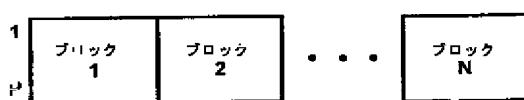
【図6】



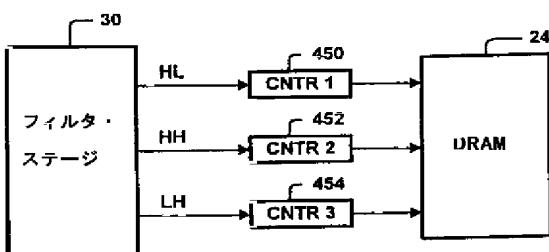
【図7】



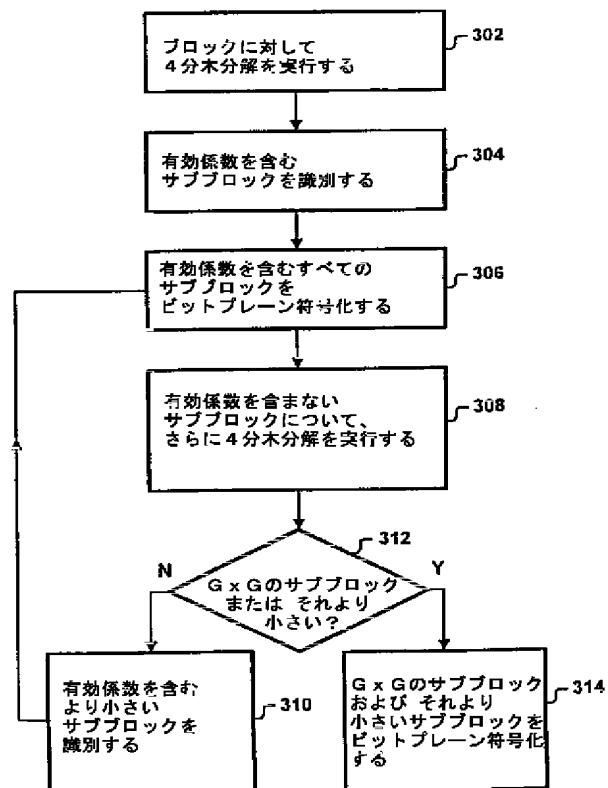
【図8】



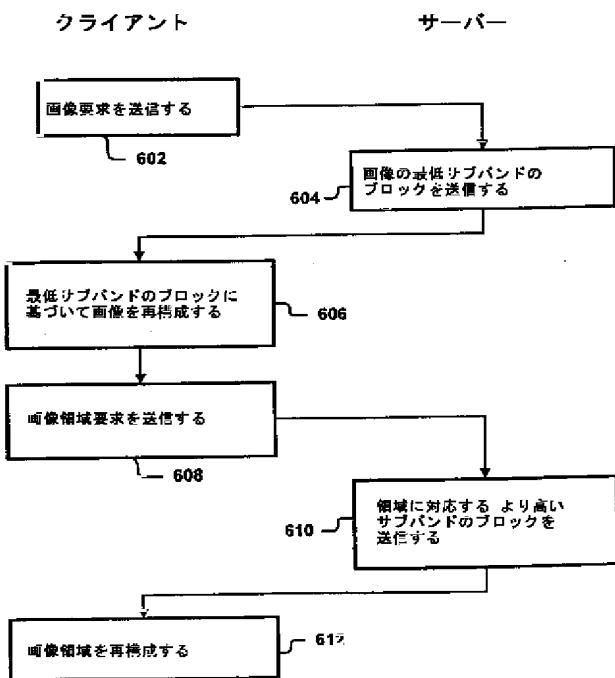
【図11】



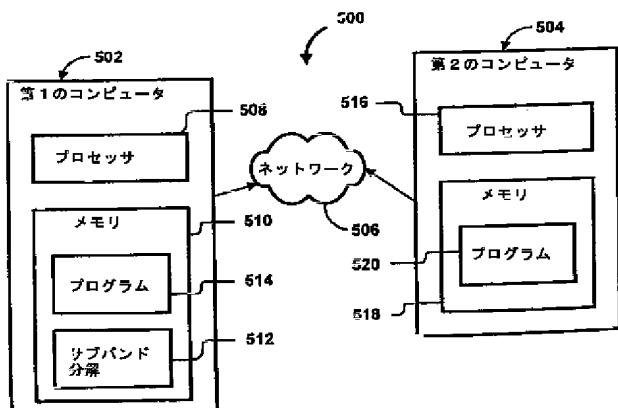
【図9】



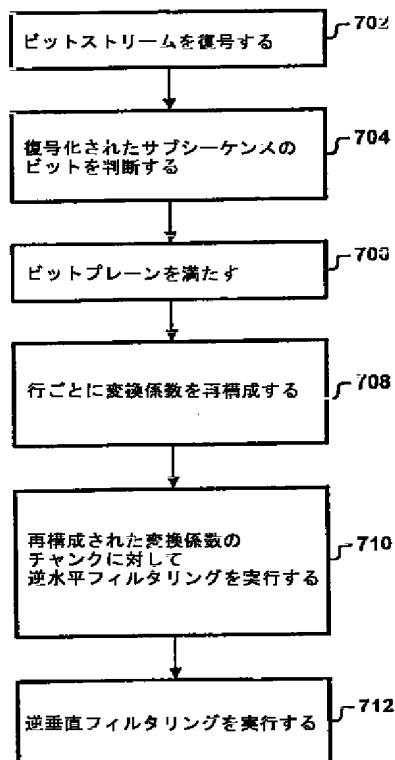
【図13】



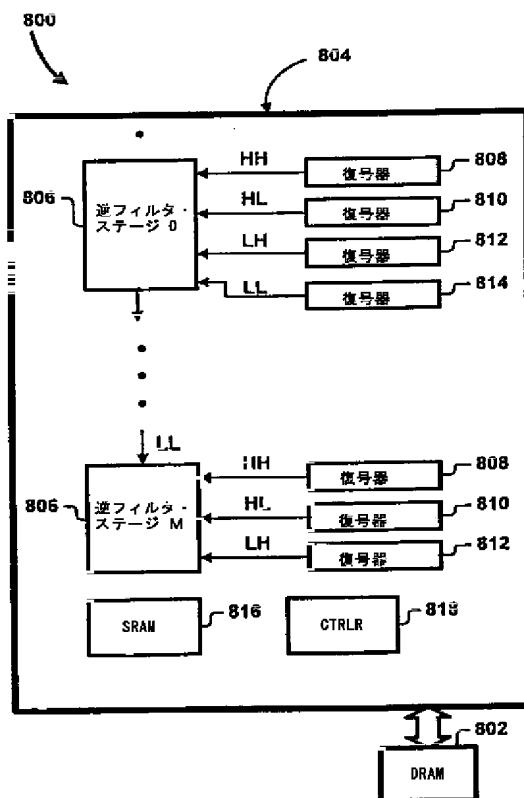
【図12】



【図14】



【図15】



---

フロントページの続き

(72)発明者 クリストス・クリサフィス  
アメリカ合衆国94040カリフォルニア州マ  
ウンテン・ビュー、イースト・エル・カミ  
ノ・レアル 870

(72)発明者 エリック・オーデントリヒ  
アメリカ合衆国94306カリフォルニア州バ  
ロ・アルト、グラント・アヴェニュー  
425、ナンバー-36